What is claimed is:

1. An apparatus for instruction-level parallelism in a processing element, comprising:

an instruction control unit;

a first instruction buffer coupled to said instruction control unit;

a second instruction buffer coupled to said instruction control unit;

a dependency counter coupled to said instruction control unit;

an execution switch coupled to said instruction control unit, said first instruction buffer, and said second instruction buffer; and

an execution unit coupled to said execution switch.

2. The apparatus of claim 1, wherein said dependency counter includes a first counter associated with the first instruction buffer and a second counter associated with the second instruction buffer.

3. The apparatus of claim 1, wherein said instruction control unit identifies instruction dependency bits in said first instruction buffer, the instruction dependency bits being associated with instructions.

4. The apparatus of claim 1, said instruction control unit generating control signals based on the dependency bits and values included in said dependency counter.

5. The apparatus of claim 4, said execution switch providing instructions from said first instruction buffer to said execution unit based on control signals from said instruction control unit.

6. The apparatus of claim 1, said execution switch providing instructions from said first instruction buffer to said execution unit based on control signals from said instruction control unit.

7. An apparatus for processing instructions in multiple threads in an execution unit, comprising:

an instruction buffer holding a first instruction and a second instruction, the first instruction being associated with a first thread, and the second instruction being associated with a second thread;

a dependency counter;

an instruction control unit coupled to said instruction buffer and said dependency counter, said instruction control unit detecting instruction dependency bits and incrementing and decrementing said dependency counter; and

an execution switch coupled to said instruction control unit and said instruction buffer, said execution switch sending instructions to the execution unit.

8. The apparatus of claim 7, wherein said dependency counter includes a first counter associated with the first thread and a second counter associated with the second thread.

9. The apparatus of claim 7, wherein said instruction buffer includes the instruction dependency bits, the instruction dependency bits being associated with instructions.

10. The apparatus of claim 7, wherein said instruction control detects dependency between the first instruction and the second thread based on dependency bits in said instruction buffer and a value of said dependency counter.

11. The apparatus of claim 7, wherein said first processing element and said second processing element is disposed within a telecommunications switch.

12. An apparatus for instruction-level parallelism, comprising:

an instruction buffer holding a first instruction and a second instruction, the first instruction being associated with a first thread, and the second instruction being associated with a second thread; and

an instruction control unit coupled to said instruction buffer, said instruction control unit detecting instruction dependency bits that indicate dependency between an instruction and one or more threads other than the thread with which the instruction is associated, and sending instructions to the execution unit to be executed.

13. The apparatus of claim 12, wherein said instruction control unit identifies instruction dependency bits in said instruction buffer, the instruction dependency bits being associated with the first instruction and the second instruction.

14. The apparatus of claim 12, wherein said instruction control detects dependency between the first instruction and the second instruction based on dependency bits in said instruction buffer.

15. A method for processing instructions in multiple threads, comprising:

receiving a first instruction associated with a first thread;

determining that execution of the first instruction depends on execution of a second instruction, the second instruction being associated with a second thread;

examining a counter associated with the first thread if said determining indicates that the first instruction depends on the execution of the second instruction;

decrementing the counter if said examining indicates that the second instruction has already been executed; and

executing the first instruction.

16. The method of claim 15, further comprising suspending the processing of the first thread until said examining indicates that the second instruction has already been executed.

17. A method for processing instructions in multiple threads, comprising:

receiving a first instruction associated with a first thread;

determining that execution of a second instruction depends on the execution of the first instruction, the second instruction being associated with a second thread;

incrementing a counter associated with the second thread if said determining indicates that execution of a second instruction depends on the execution of the first instruction; and

executing the first instruction.

18. The method of claim 17, further comprising suspending the processing of the second thread if the counter associated with the second thread does not exceed a threshold.

19. A method for processing instructions in multiple threads, comprising:

receiving a first instruction associated with a first thread;

determining that the first instruction identifies a second thread;

incrementing a counter associated with the second thread if said first instruction identifies the second thread;

loading a second instruction associated with a second thread; and

processing the second instruction in a manner related to the value of the counter associated with the second thread.

20. The method of claim 19, further comprising suspending the processing the second thread if the counter indicates that a dependent thread has not been executed.

21. The method of claim 19, further comprising executing the second instruction if the counter indicates that said first instruction has been executed.

22. A method for processing instructions in multiple threads, comprising:

receiving a first instruction associated with a first thread;

determining the first instruction identifies a second thread;

examining a counter associated with the second thread;

decrementing the counter if said examining indicates a non-zero value; and

executing the first instruction.

23. A method for processing instructions in multiple threads, comprising:

loading a first instruction associated with a first thread;

detecting dependency between the first instruction and a second instruction associated with a second thread based on dependency bits in an instruction buffer and the value of a dependency counter.

24. An apparatus for processing instructions in multiple threads, comprising:

an instruction buffer configured to hold a first instruction and a second instruction, the first instruction including a dependency indicator and being associated with a first thread, and the second instruction including a dependency indicator and being associated with a second thread;

an instruction control unit coupled to said instruction buffer;

a dependency counter coupled to said instruction control unit, said dependency counter associated with the first thread;

said instruction control unit configured to detect the dependency indicators and change the value of said dependency counter in response to detecting the dependency indicators; and

said instruction control unit configured to disallow execution of the first instruction if said dependency counter includes a value less than a threshold value.

25. The apparatus of claim 24, wherein said instruction control unit is configured to determine that the dependency indicator included in the first instruction indicates that the second thread includes an instruction on which the first instruction depends.

26. The apparatus of claim 24, wherein the dependency indicator included in the first instruction is a depends bit.

27. The apparatus of claim 24, wherein said instruction control unit is configured to determine that the dependency indicator included in the second instruction indicates that the first thread includes an instruction that is dependent on the second instruction.

28. The apparatus of claim 24, wherein the dependency indicator included in the second instruction is a tells bit.

29. The apparatus of claim 24, wherein said instruction control unit is configured to increment said dependency counter in response to detecting the dependency indicator included in the second instruction.

30. The apparatus of claim 24, wherein said instruction control unit is configured to decrement said dependency counter in response to detecting the dependency indicator included in the first instruction.